

The Devil Is In The Details. . . And The Data — Tutorial On Preparing Data for Multi-state Modelling

Enoch Yi-Tung Chen

Department of Medical Epidemiology and Biostatistics, Karolinska Institutet

The 2022 Northern European Stata Conference
12 Oct 2022

<https://github.com/enochytchen/NordicStata2022>

Illness-death model

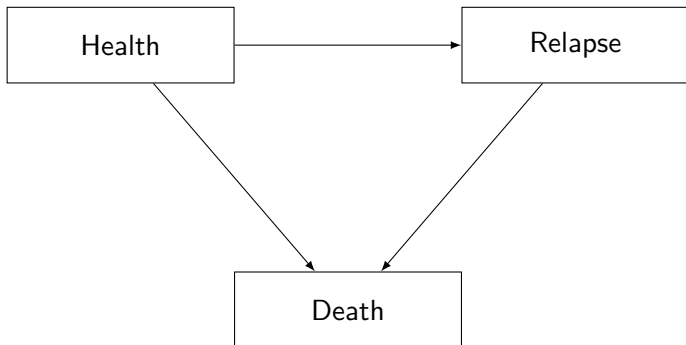


Figure: Illness-death model

Illness-death model

Example data from Crowther2017 (1)

```
. use http://fmwww.bc.edu/repec/bocode/m/multistate\_example, clear  
(Rotterdam breast cancer data, truncated at 10 years)  
  
. // List one patient to see the variables  
. list pid rf rfi os osi if pid == 1 | pid == 1371, sepby(pid) noobs
```

```
+-----+  
| pid    rf    rfi    os    osi |  
+-----+  
|    1   59.1    0   59.1   alive |  
+-----+  
| 1371   16.6    1   24.3  deceased |  
+-----+
```

Illness-death model

Transition matrix for Illness-death model

```
. matrix tmat = (.,1,2\ .,.,3\.,.,.)  
. matrix colnames tmat = to:Health to:Relapse to:Death  
. matrix rownames tmat = from:Health from:Relapse from:Death  
. matrix list tmat
```

```
tmat[3,3]
```

	to: Health	to: Relapse	to: Death
from:Health	.	1	2
from:Relapse	.	.	3
from:Death	.	.	.

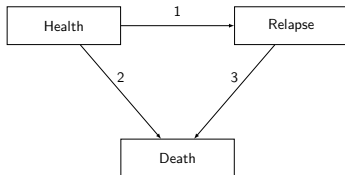


Figure: Illness-death model

multistate::msset

```
. msset, id(pid) states(rfi osi) times(rf os) transm(tmat)
. list pid rf rfi os osi _trans _start _stop if pid == 1 | pid ==1371, sepby(pid)
```

pid	rf	rfi	os	osi	_trans	_start	_stop
1	59.1	0	59.1	alive	1	0	59.104721
1	59.1	0	59.1	alive	2	0	59.104721
1371	16.6	1	24.3	deceased	1	0	16.558521
1371	16.6	1	24.3	deceased	2	0	16.558521
1371	16.6	1	24.3	deceased	3	16.558521	24.344969

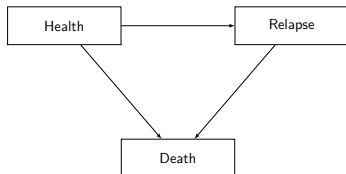


Figure: Illness-death model

Reversible illness-death model

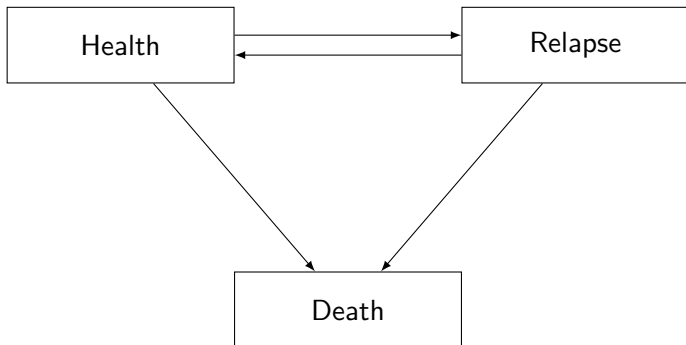


Figure: Reversible illness-death model

Example data from Crowther2017

```
. use http://fmwww.bc.edu/repec/bocode/m/multistate_example, clear
(Rotterdam breast cancer data, truncated at 10 years)

. // Assume recovery indicator and recovery time
. set seed 12345

. // Recovery indicator
. gen rei = cond(runiform() < 0.5, 0 , 1) if rfi == 1 & rf!= os
(1,464 missing values generated)

. // Recovery
. gen re = runiform(rf, os) if rei == 1
(2,243 missing values generated)

. save multistate_example_temp.dta, replace

. // List one patient to see the variables
. list pid rf rfi re rei os osi if pid == 2778 , sepby(pid) noobs
```

```
+-----+
| pid    rf    rfi      re    rei      os    osi |
+-----+
| 2778   40.3    1    53.36185    1    114.0  alive |
+-----+
```

Transition matrix for reversible illness-death model

```
. matrix rtmat = (.,1,2\ 3,.,4\.,.,.)  
. matrix colnames rtmat = to:Health to:Relapse to:Death  
. matrix rownames rtmat = from:Health from:Relapse from:Death  
. matrix list rtmat
```

```
rtmat[3,3]
```

	to: Health	to: Relapse	to: Death
from:Health	.	1	2
from:Relapse	3	.	4
from:Death	.	.	.

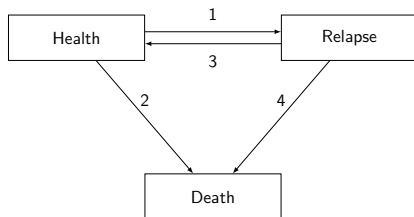


Figure: Reversible illness-death model

Common mistake 1

```
. msset, id(pid) states(rfi osi rei) times(rf os re) transm(rtmat)
All elements of the lower triangle of transmatrix() must be coded missing = .
```

```
. matrix list rtmat
rtmat[3,3]
```

	to:	to:	to:
	Health	Relapse	Death
from:Health	.	1	2
from:Relapse	3	.	4
from:Death	.	.	.

Common mistake 2

```
. msset, id(pid) states(rfi osi rei) times(rf os re)
. list pid rf rfi re rei os osi _trans _start _stop _status if pid == 2778
```

	pid	rf	rfi	re	rei	os	osi	_trans	_start	_stop	_status
11725.	2778	40.3	1	53.36185	1	114.0	alive	1	0	40.279263	1
11726.	2778	40.3	1	53.36185	1	114.0	alive	2	0	40.279263	0
11727.	2778	40.3	1	53.36185	1	114.0	alive	3	0	40.279263	0
11728.	2778	40.3	1	53.36185	1	114.0	alive	4	40.279263	53.361855	0
11729.	2778	40.3	1	53.36185	1	114.0	alive	5	40.279263	53.361855	1

Set up a multi-state structured data

msset created the following variables

```
/*
_from          float    %9.0g  Starting state
_to            float    %9.0g  Receiving state
_status        byte     %8.0g  Event (transition) indicator
_start         double   %10.0g  Starting time for each transition
_stop          double   %10.0g  Stopping time for each transition
_flag          byte     %8.0g  Data modified
_trans         float    %9.0g  Transition number
_trans1        byte     %8.0g  _trans== 1.0000
_trans2        byte     %8.0g  _trans== 2.0000
_trans3        byte     %8.0g  _trans== 3.0000
*/

// Generate variables
gen _from = .
gen _to = .
gen _start = .
gen _stop = .
gen _status = .
```

Set up a multi-state structured data

Target: make wide-format data into long-format

```
. // List one patient to see the variables
. list pid rf rfi re rei os osi if pid == 2778 , sepby(pid) noobs
+-----+
| pid   rf   rfi       re  rei       os   osi |
+-----+
| 2778  40.3   1   53.36185   1   114.0  alive |
+-----+

. matrix list rtmat
rtmat[3,3]
           to:      to:      to:
           Health  Relapse  Death
from:Health  .         1         2
from:Relapse 3         .         4
from:Death   .         .         .
```

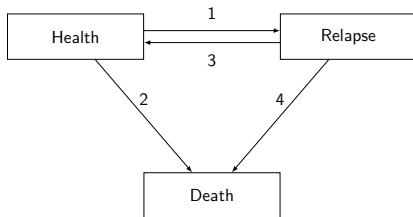


Figure: Reversible illness-death model

Set up a multi-state structured data

```
// Make 4 duplicates for each patient to define transitions
expand 4

// Manually make msset format
bysort pid: gen _trans = _n
// Generate _episode for potential recurrent events after recovery
gen _episode = 1
expand 2 if (_tran == 1 | _tran == 2) & rei == 1, gen(du)
replace _episode = 2 if du == 1
drop du
```

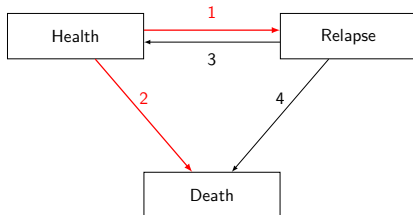


Figure: Reversible illness-death model

Set up a multi-state structured data

```
// Check the duplicates were done correctly
list pid rf rfi re rei os osi _trans _episode if pid == 2778 , sepby(pid) noobs
+-----+
| pid    rf    rfi      re  rei    os   osi  _trans  _episode |
+-----+-----+
| 2778   40.3   1    53.36185   1   114.0  alive    1        1 |
| 2778   40.3   1    53.36185   1   114.0  alive    2        1 |
| 2778   40.3   1    53.36185   1   114.0  alive    3        1 |
| 2778   40.3   1    53.36185   1   114.0  alive    4        1 |
| 2778   40.3   1    53.36185   1   114.0  alive    1        2 |
| 2778   40.3   1    53.36185   1   114.0  alive    2        2 |
+-----+-----+
```

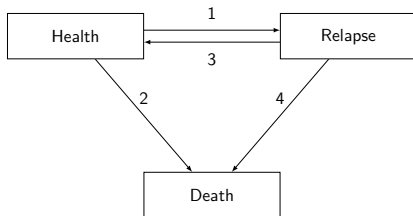


Figure: Reversible illness-death model

Set up a multi-state structured data

Specify `_from` `_to`

```
. matrix list rtmat

rtmat[3,3]
      to:      to:      to:
      Health  Relapse  Death
from:Health   .        1        2
from:Relapse  3        .        4
from:Death    .        .        .

replace _from = 1 if _trans == 1 | _trans == 2
replace _from = 2 if _trans == 3 | _trans == 4

replace _to = 1 if _trans == 3
replace _to = 2 if _trans == 1
replace _to = 3 if _trans == 2 | _trans == 4
```

Set up a multi-state structured data

Specify `_start` `_stop`

```
local condition "(_trans == 1 | _trans == 2) & _episode == 1"  
replace _start = 0 if 'condition' // T0 is 0  
replace _stop = min(rf,os) if 'condition' // Replase, death/censoring,  
// whichever happens first  
replace _stop = 120 if _stop == . & 'condition' // Censor everyone after 120 mos
```

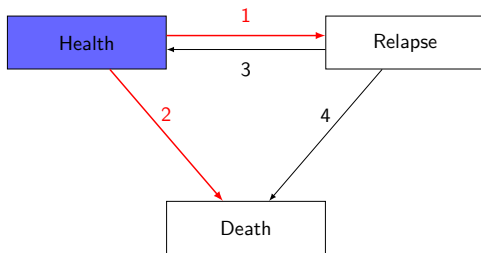


Figure: Reversible illness-death model

Set up a multi-state structured data

Specify `_start` `_stop`

```
local condition "(_trans == 3 | _trans == 4)"
replace _start = rf if 'condition' // T0 is time since relapse
replace _stop = min(re,os) if 'condition' // Recovery, death/censoring,
// whichever happens
replace _stop = 120 if _stop == . & 'condition' // Censor everyone after 120 mos
```

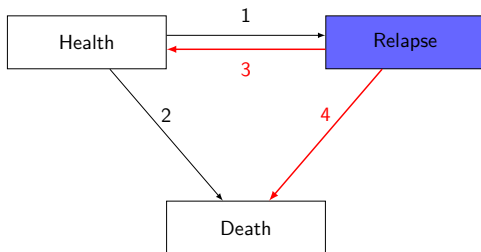


Figure: Reversible illness-death model

Set up a multi-state structured data

Specify `_start` `_stop`

```
local condition "(_trans == 1 | _trans == 2) & _episode == 2"  
replace _start = re if 'condition' // T0 is time since recovery  
replace _stop = os if 'condition' // Death/censoring,  
// whichever happens  
replace _stop = 120 if _stop == . & 'condition' // Censor everyone after 120 mos
```

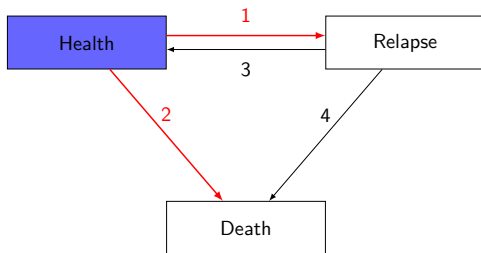


Figure: Reversible illness-death model

Set up a multi-state structured data

Specify `_status`

```
// _trans == 1 & _episode == 1
replace _status = 1 if _trans == 1 & _episode == 1 & ///
    rfi == 1 & min(rf,os) == rf //
    // Relapse as an event and happens first
replace _status = 0 if _trans == 1 & _episode == 1 & _status != 1

// _trans == 2 & _episode == 1
replace _status = 1 if _trans == 2 & _episode == 1 & ///
    osi == 1 & rfi == 0 //
    // Death/censoring as an event and relapse never happens
replace _status = 0 if _trans == 2 & _episode == 1 & _status != 1
```

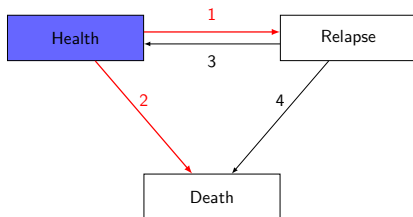


Figure: Reversible illness-death model

Specify _status

```
// _trans == 3
replace _status = 1 if _trans == 3 & rfi == 1 & /// Relapse has happened
    rei == 1 // Recovery as an event
replace _status = 0 if _trans == 3 & rfi == 1 & /// Relapse has happened
    _status != 1

// _trans == 4
replace _status = 1 if _trans == 4 & rfi == 1 & /// Relapse has happened
    osi == 1 & rei == 0 //
    // Death/censoring as an event and happens first
replace _status = 0 if _trans == 4 & rfi == 1 & /// Relapse has happened
    _status != 1
```

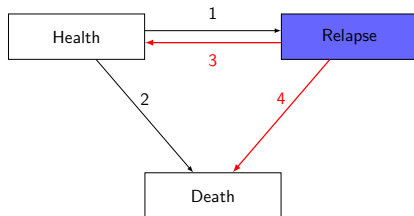


Figure: Reversible illness-death model

Specify _status

```
// _trans == 1 & _episode == 2
replace _status = 1 if _trans == 1 & _episode == 2 & ///
    rfi == 1 & rei == 1 & /// Relapse and recovery have happened
    min(rf,re) == re // Recovery happens first, impossible here.
replace _status = 0 if _trans == 1 & _episode == 2 & ///
    rfi == 1 & rei == 1 & /// Relapse and recovery have happened
    _status != 1
// _trans == 2 & _episode == 2
replace _status = 1 if _trans == 2 & _episode == 2 & ///
    rfi == 1 & rei == 1 & /// Relapse and recovery have happened
    osi == 1 & min(re,os) == re // Death/censoring as an event a
replace _status = 0 if _trans == 2 & _episode == 2 & ///
    rfi == 1 & rei == 1 & _status != 1
```

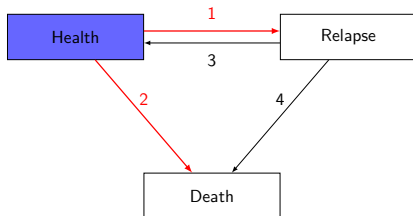


Figure: Reversible illness-death model

Set up a multi-state structured data

Check

```
// List those who are not at risk in each transition
// There shouldn't be any missing tho
// If there is, it means there's something wrong

list pid _start _stop _from _to _status _trans ///
      if _start == . | _stop == . | _status == .
```

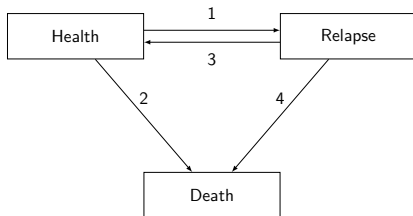


Figure: Reversible illness-death model

Set up a multi-state structured data

Check

```
. list pid rf rfi re rei os osi _trans _episode _start _stop _status if pid == 2846
```

	pid	rf	rfi	re	rei	os	osi	_trans	_episode	_start	_stop	_status
11281.	2846	24.3	1	26.14	1	72.7	deceased	1	1	0	24.34	1
11282.	2846	24.3	1	26.14	1	72.7	deceased	2	1	0	24.34	0
11283.	2846	24.3	1	26.14	1	72.7	deceased	3	1	24.34	26.14	1
11284.	2846	24.3	1	26.14	1	72.7	deceased	4	1	24.34	26.14	0
13397.	2846	24.3	1	26.14	1	72.7	deceased	1	2	26.14	72.73	0
13398.	2846	24.3	1	26.14	1	72.7	deceased	2	2	26.14	72.73	1

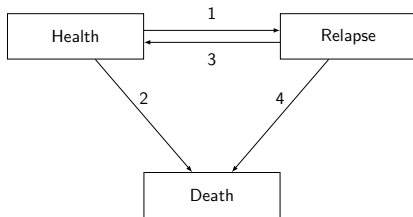


Figure: Reversible illness-death model

Check

```
. tab _tr_epi _status
```

_tr_epi	_status		Total
	0	1	
1_1	1,464	1,518	2,982
1_2	739	0	739
2_1	2,787	195	2,982
2_2	217	522	739
3_1	779	739	1,518
4_1	963	555	1,518
Total	6,949	3,529	10,478

```
preserve
```

```
. use multistate_example_temp, clear  
(Rotterdam breast cancer data, truncated at 10 years)
```

```
. tab rfi
```

Relapse indicator	Freq.	Percent	Cum.
0	1,464	49.09	49.09
1	1,518	50.91	100.00
Total	2,982	100.00	

```
restore
```

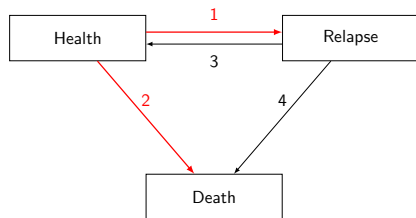


Figure: Reversible illness-death model

Check

```
. tab _tr_epi _status
```

_tr_epi	0	1	Total
1_1	1,464	1,518	2,982
1_2	739	0	739
2_1	2,787	195	2,982
2_2	217	522	739
3_1	779	739	1,518
4_1	963	555	1,518
Total	6,949	3,529	10,478

```
preserve
```

```
. use multistate_example_temp, clear  
(Rotterdam breast cancer data, truncated at 10 years)
```

```
. tab rei
```

rei	Freq.	Percent	Cum.
0	779	51.32	51.32
1	739	48.68	100.00
Total	1,518	100.00	

```
restore
```

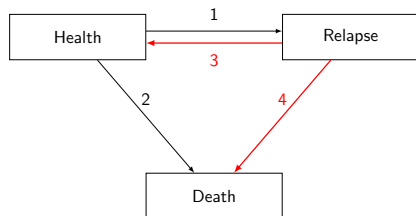


Figure: Reversible illness-death model

Check

```
. tab _tr_epi _status
```

_tr_epi	0	1	Total
1_1	1,464	1,518	2,982
1_2	739	0	739
2_1	2,787	195	2,982
2_2	217	522	739
3_1	779	739	1,518
4_1	963	555	1,518
Total	6,949	3,529	10,478

```
preserve
```

```
. use multistate_example_temp, clear  
(Rotterdam breast cancer data, truncated at 10 years)
```

```
. tab osi
```

Overall survival	Freq.	Percent	Cum.
alive	1,710	57.34	57.34
deceased	1,272	42.66	100.00
Total	2,982	100.00	

```
. restore
```

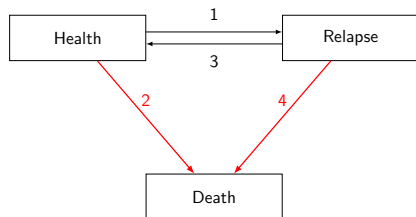


Figure: Reversible illness-death model

Summary

1. Always put the diagram aside
2. Thinking about competing risk
(What has happened? What may happen next? Risk set?)
3. No error \neq correct data. The devil is in the details and the data!

Summary

1. Always put the diagram aside
2. Thinking about competing risk
(What has happened? What may happen next? Risk set?)
3. No error \neq correct data. The devil is in the details and the data!
4. Then...happy hour for multi-state modelling!

Acknowledgements

Nikolaos Skourlis (Karolinska Institutet)

References

1. Crowther MJ, Lambert PC. Parametric multi-state survival models: flexible modelling allowing transition-specific distributions with application to estimating clinically useful measures of effect differences. *Statistics in Medicine* 2017;**36**:4719–4742.

Appendix

Slides and syntax of this presentation can be found at:
<https://github.com/enochytchen/NordicStata2022>